

REMARKS/ARGUMENTS

The Examiner rejected claims 1-20 as obvious (35 U.S.C. §103(a)) over Bates (U.S. Patent No. 6,977,927) in view of Phillips (U.S. Patent No. 5,321,828). Applicants traverse.

Amended claim 1 recites a storage area network (SAN) including a first and second digital data processors executing a first and second operating systems, respectively, in communication with one or more storage devices, comprising: a first platform-specific process executing on the first digital data processor; a second platform-specific process executing on the second digital data processor, wherein the second operating system is different from the first operating system; a first common platform-independent process executing on the first digital data processor, wherein the first common platform-independent process invokes and communicates with a first command line interface of the first operating system to effect execution of the first platform-specific process via command line parameters; and a second common platform-independent process executing on the second digital data processor, wherein the second common platform-independent process invokes and communicates with a second command line interface of the second operating system to effect execution of the second platform-specific process via command line parameters.

Applicants amended claim 1 to clarify that the second common platform independent process invokes and communicates with a second (not first) command line interface of the second (not first) operating system. The additional requirements of these claims are disclosed on at least pgs. 42-43 and 191-193 of the Application, and commensurate with the requirements of other dependent claims 15 and 21.

The Examiner cited col. 23, lines 50-67 of Phillips as teaching the claim requirements of first and second common platform independent processes executing on first and second processors, wherein the first and second common platform independent processes invoke and communicate with a first and second command line interfaces, respectively, to effect execution of first and second platform specific processes, respectively. (Third Office Action, pg. 3) Applicants traverse.

The cited col. 23 of Phillips discusses GDB, a standard debugger that runs on the UNIX operating system. The source code of GDB is converted to a format compatible with a Microsoft "C" compiler running on DOS. Certain standard functions are altered to call their equivalents in "C". The GDB DLL retains its command line interface and does not allow Windows

applications to link to its modules. Thus, the debugger maintains its quality of remaining a stand alone executable module, but can communicate with Windows applications via ASCII strings.

The cited col. 23 discusses how to convert a debugger (GDB) to run in DOS to be compatible with a “C” compiler. Nowhere does the cited col. 23 teach or suggest the claim requirements of common platform independent processes invoking and communication with first and second command line interfaces of the first and second operating systems to effect execution of first and second platform specific processes via command line parameters. The cited col. 23 mentions that the debugger (GDB) maintains a command line interface. However, there is no teaching or suggestion in the cited col. 23 of platform independent processes effecting execution of first and second platform specific processes via command line interfaces. Instead, the cited col. 23 mentions that the debugger maintains a command line interface, not uses one.

Moreover, the cited col. 23 teaches away from the claim requirement of platform independent processes effecting execution of first and second platform specific processes via command line interfaces. The cited col. 23 mentions that the debugger “GDB retains its command line interface and does not allow WINDOWS applications to directly link to its modules.” Phillips further mentions that the “source level debugger 22 is preferably invoked with DOS command line or UNIX shell command”. (col. 24, lines 18-30). Thus, the cited command line interface is used to control the debugger, which the Examiner likens to the common platform independent processor. There is no teaching in the cited Phillips that the debugger use a command line interface of the operating system to invoke and effect execution of platform specific processes, such as Windows applications. Instead, the cited col. 23 mentions that the debugger communicates with Windows applications via ASCII strings. Thus, the cited col. 23 teaches away from having platform independent processes use command line interfaces of the operating system to effect execution of platform specific processes because according to the cited col. 23, the command line interface is used to allow communication with the debugger.

The Examiner further cited FIG. 7, element 704 of Bates in rejecting claim 1. (Third Office Action, pg. 3) Applicants submit that the cited element 704 fails to teach the claim requirements of first and second platform independent processes using command line interfaces of the first and second operating systems, respectively, to effect execution of first and second platform specific processes.

The cited FIG. 7 shows a SAN with computers having different operating systems, including element 704, communicating with a storage. Element 704 is the Unix server. (Bates, col. 12, line 65 to col. 13, line 8). Nowhere does the cited FIG. 7 anywhere teach or suggest the claim requirements of first and second platform independent processes using command line interfaces of the first and second operating systems, respectively, to effect execution of first and second platform specific processes. Instead, the element 704 shows a Unix server in communication with storage allocator in a SAN environment.

The Examiner further cited col. 13, line 29 to col. 14, line 60 of Bates with respect to claim 1. (Third Office Action, pg. 4) The cited cols. 13-14 of Bates mentions that a storage allocator maps or masks available storage space to present to hosts. The cited cols. 13-14 further mentions virtual LUN partitions and storage security. Each host, having different operating systems, has access to separate non-overlapping physical LUNs. The storage allocator may be controlled by a user interface to manually configure the allocation of storage.

Nowhere do the cited cols. 13-14 of Bates anywhere teach or suggest the claim requirements of first and second platform independent processes using command line interfaces of the first and second operating systems, respectively, to effect execution of first and second platform specific processes. Instead, the cited cols. 13-14 discuss a storage allocator that allocates storage to different hosts.

The Examiner further cited col. 3, lines 46-67 and col. 8, lines 52-53 of Bates with respect to claim 1. (Third Office Action, pg. 4) The cited col. 3 discusses servers that process requests for data and applications in a network that output read and write requests to a storage allocator. The cited col. 8 discusses an interface between SAN servers and storage. Nowhere do the cited cols. 3 and 8 of Bates anywhere teach or suggest the claim requirements of first and second platform independent processes using command line interfaces of the first and second operating systems, respectively, to effect execution of first and second platform specific processes. Instead, the cited col. 3 discusses how servers access storage through a storage allocator.

Yet further, Applicants submit that it is inappropriate to combine Bates and Phillips because there is no suggestion in Phillips, which concerns a debugger, that its debugger be used in a SAN, such as Phillips, having a storage allocator to provide host servers access to servers.

Further, there is no suggestion in Bates that its storage allocator be used with debuggers as discussed in Phillips.

Accordingly, claim 1 is patentable over the cited Phillips and Bates because the cited combination, alone or in combination, does not teach or suggest all the claim requirements.

Claims 3-14 are patentable over the cited art because they depend from claim 1, which is patentable over the cited art for the reasons discussed above. Moreover, the following dependent claims provide further grounds of distinction over the cited art.

Claim 4 depends from claim 1 and further requires a manager in communication with the first and second common platform-independent process to transmit requests thereto for information regarding one or more components of the SAN.

The Examiner cited the previously discussed cols 3, 8 and 13-14 of Bates and col. 23 of Phillips as teaching the requirements of claim 4. (Third Office Action, pgs. 5-6) Applicants traverse.

The cited Bates discusses a storage allocator that provides information on LUNs and storage to use to the different storage servers having different operating systems. However, nowhere does the cited Bates anywhere teach or suggest that the servers in Bates comprise platform independent processes, such that the storage allocator transmits requests to platform independent processes for information on components of the SAN. Instead, the cited Bates discusses how the storage allocator provides LUN assignments to the servers and handles read and write requests to the storage.

The cited Phillips discusses a debugger. Nowhere does the cited Phillips anywhere teach or suggest a manager in communication with first and second common platform-independent process to transmit requests thereto for information regarding one or more components of the SAN.

Accordingly, claim 4 provides additional grounds of patentability over the cited art because the additional requirements of these claims are not disclosed in the cited art.

Claim 6 depends from claim 5 and further requires that the invoked first and second platform specific processes gather information regarding one or more SAN components and transmit the information to the Standard Output/Error of their respective first and second digital data processors.

The Examiner cited col. 3, lines 37-67 of Bates and the previously discussed col. 8 of Bates and col. 23 of Phillips as teaching the additional requirements of claim 6. (Third Office Action, pg. 7) Applicants traverse.

The cited col. 3 discusses the above discussed network of servers with different operating systems, a storage allocator and storage, where the storage allocator receives read and write requests from the server to determine the storage locations for the request. The discussed cited col. 23 of Phillips discusses a debugger that has a command line interface through which it may be invoked.) Nowhere do the cited Phillips and Bates anywhere teach or suggest separate first and second platform specific processes executing on different processors having different operating systems gathering information on SAN components and transmit the gathered information to the standard output/error.

Accordingly, claim 6 provides additional grounds of patentability over the cited art because the additional requirements of these claims are not disclosed in the cited art.

Claims 7 and 8, which depend from claim 6, also provide additional grounds of patentability over the cited art for requiring further operations with respect to the information transmitted to the standard output/error of the first and second digital data processors.

Claim 9 recites that the manager comprises a query engine for transmitting the requests to the first and second common platform independent processes.

The Examiner cited col. 15, lines 5-22 of Bates as teaching a query and the above discussed col. 3, lines 46-67 of Bates as teaching multiple processors and platform specific operations. (Third Office Action, pg. 9) Applicants traverse.

The cited col. 15 mentions that the storage allocator is implemented in a platform independent language, such as Java. The cited col. 3 discusses how the storage allocator provides access to storage to servers having different operating systems. Although the cited Bates mentions that the storage allocator is in a platform independent language, nowhere does the cited Bates (or Phillips) anywhere teach or suggest that the storage allocator has a query engine for transmitting requests to first and second common platform independent processes on different processors having different operating systems as claimed. Further, FIG. 7 shows the storage allocator separate from the servers. The claims require that first and second platform independent processes receiving requests from a query engine. Nowhere does the cited Bates anywhere teach or suggest that the servers 702, 704, and 706 in FIG. 7 have platform

independent processes for receiving requests for information on SAN components from a query engine in first and second. Instead, the cited Bates discusses how the storage allocator receives read and write requests from the servers and is written in a platform independent computer language.

Accordingly, claim 9 provides additional grounds of patentability over the cited art because the additional requirements of these claims are not disclosed in the cited art.

Claims 10-14 provide additional grounds of patentability over the cited art because they provide further details on the manager query engine.

Claim 15 was amended to clarify that the second command line interface is invoked to effect execution of the second (not the first) platform specific process. Claim 15 is patentable over the cited art for the reasons discussed with respect to claim 1.

Claims 16-20 are patentable over the cited art because they depend from claim 15, which is patentable over the cited art for the reasons discussed above. Further, claims 16-19 include similar limitations to claims 6, 7, 9, and 10, and thus provide additional grounds of patentability over the cited art for the reasons discussed with respect to claims 6, 7, 9, and 10.

Claim 21 substantially includes the requirements of amended claim 1. Applicants submit that claim 21 is patentable over the cited art for the reasons discussed with respect to amended claim 1. Claims 23-26 substantially include the requirements of claims 4, 5, 6, and 9 in computer readable media form and thus are patentable over the cited art for the reasons discussed above with respect to claims 4, 5, 6, and 9.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1, 3-21, and 23-26 are patentable over the art of record. Applicants submit that no additional fee is needed. Nonetheless, should any additional fees be required, including any additional extension of time fees, please charge Deposit Account No. 09-0466.

\\
\\
\\
\\
\\

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: April 25, 2006

By: /David Victor/

David W. Victor
Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984